5-2 Milestone Four: Databases

Maria Herring

CS 499 Computer Science Capstone

Instructor: Brooke Goggin

February 9, 2024

Revised February 10, 2024

* 1. Milestone Four: Databases

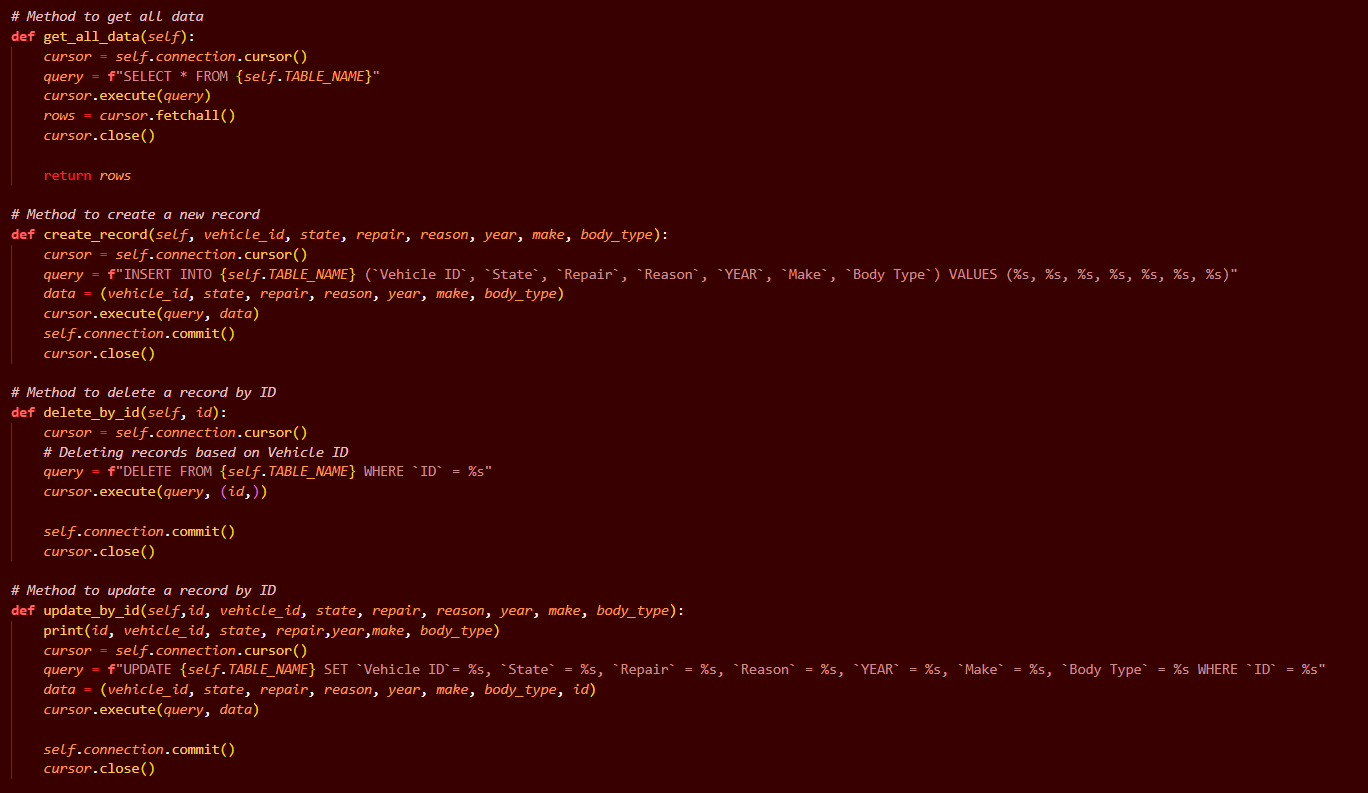
The artifact I chose to enhance in the Databases category is a from DAD-220 (Introduction into Structure Databases), which was completed in the fall of 2022. The artifact was originally used in a virtual lab, using a Linux platform. The artifact is a CSV data file used in the course to write queries. The file contains vehicle, damage, and repair statistics for a fleet of vehicles. I selected this artifact so that I could use the information contained in the file to develop an application with create, read, update, and delete functionality to manage the data.

Developing a user interface that allows a user to manipulate the data imported into MySQL demonstrates my understanding of databases. The project was developed in the Python programming language, and MySQL was used to store and manage the database. The MySQL Workbench tool was used to create the database and table. Visual Studio Code was used as the code editor for this project. The use and integration of these tools demonstrates my ability to develop a secure, fully functional database application.

For this project, a new user and password were created from the MYSQL root account using a MySQL shell. Using the MySQL Workbench, a new connection was launched to create the database. Once the database, named fleet, was connected, I used the appropriate SQL syntax to create a table called Records to store the data. The existing CSV data file was imported into the fleet database. To develop the application to interact with the database a user class in Python was developed containing the database initialization, connection, and methods for the CRUD functionality. A main Python file was created using Tkinter, a standard Python interface for creating graphical user interfaces (GUI). The main Python file contains functions, which interact with the user class to delete, create, and update data.

The Tkinter user interface to display the fleet records was created using a treeview widget (trv). Columns for each table attribute were created along with a scroll bar (vsb) which allows the user to scroll through the records table. Buttons to exit, create, update, and delete were added. The update and create functions contain the logic to update and create in the database. Frames and labels that were created with widgets for editing and creating the table attributes. Implementing the application structure and logic demonstrates skills align with course objectives which demonstrate an ability to use well-founded and innovative techniques, skills, and tools in computing practices for the purpose of implementing computer solutions that deliver value and accomplish industry-specific goals.

Figure #1: CRUD functionality.



. The mechanisms to initiate CRUD functionality are complete and fully useable to manipulate and interact with the data. The use of descriptive variable names makes the program code clear, readable, and easy to understand. The naming conventions for methods and variables are consistent and descriptive. Specifically, the structure, programming logic, and design demonstrate my understanding of the Python programming language. The methods in the user Python module contain CRUD functionality that could be reusable in other applications. These skills align with course objectives which demonstrate an ability to use well-founded and innovative techniques, skills, and tools in computing practices for the purpose of implementing computer solutions that deliver value and accomplish industry-specific goals.

Figure #2: Create function with error exception.

A screenshot of a computer program

Description automatically generated

The program was developed using error checking engineering practices that allow the application to fail gracefully, exit without crashing, and help to make the program more secure. To help prevent SQL injection, the application was developed using practices such as input filtering and query parameterization. Adherence to secure coding best practices was used to develop the project, which will help ensure that the application functions properly, is secure, and maintainable. These skills demonstrate and align with course outcomes that develop a security mindset that anticipates adversarial exploits in software architecture and designs to expose potential vulnerabilities, mitigate design flaws, and ensure privacy and enhanced security of data and resources.

While enhancing this artifact, I re-familiarized myself with the syntax of the SQL programming language. I learned how to use the MySQL Workbench development tool while creating this artifact as. Previously, I used the command line for database projects. I had previously used the Tkinter graphical user interface (GUI) toolkit to build smaller projects, but I expanded my knowledge during the development of this artifact. Implementing the widget features of Tkinter, such as the tree view, allowed the data to be displayed in a table, which is a visually appealing way to interact with the data.

This artifact was developed using a file from my previous course, so it was a fun challenge to create the layout to incorporate all the aspects of the program needed to make it function. Installing the import Python package (pymysql) was surprisingly tricky, but once I used the proper connection method, it functioned properly. Through this enhancement, I increased my comfort level with the Python programming language, with MySQL, and I increased my ability to properly develop a database application.

Figure #3: Application Home Page

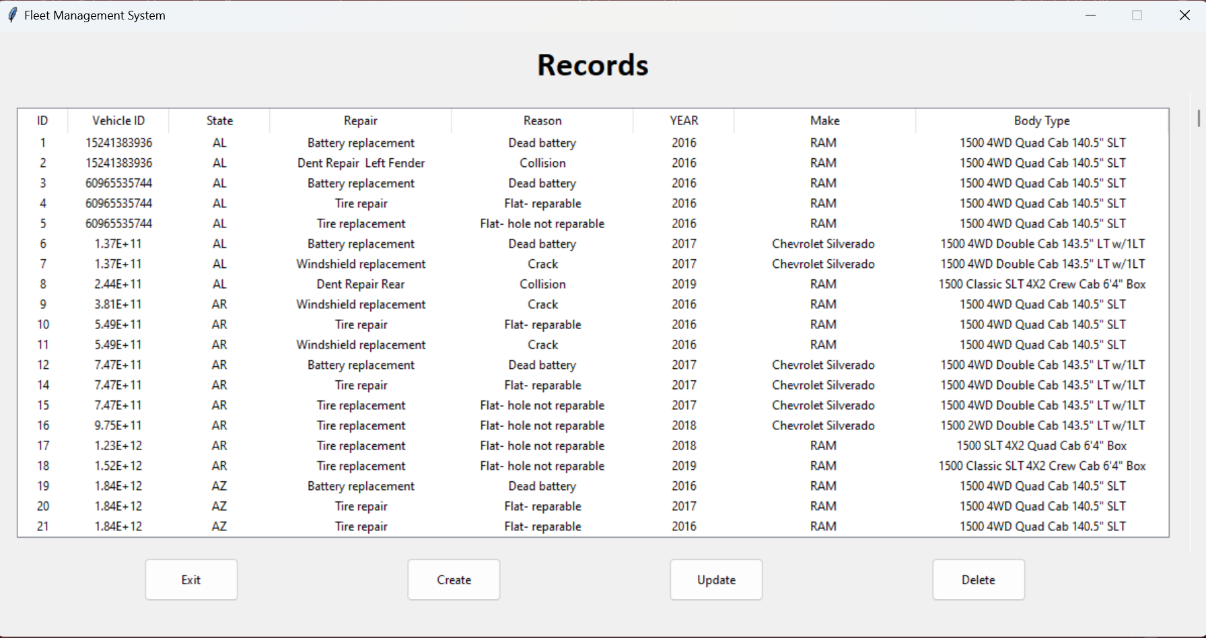


Figure #4: Add a new record.

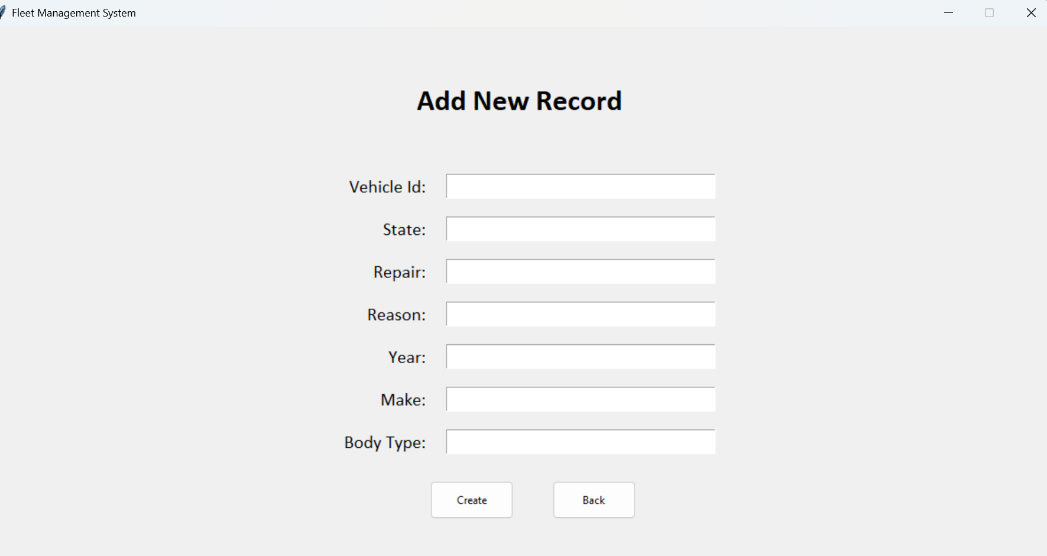


Figure #5: Update a record.

A screenshot of a computer

Description automatically generated

Figure #4: Delete a record.

A screenshot of a computer

Description automatically generated